

10 METHOD AND SYSTEM FOR PROVISIONING SPECIFICATION
SUBSETS FOR STANDARDS-BASED COMMUNICATION NETWORK
DEVICES

CROSS REFERENCE TO RELATED APPLICATION

15 This application claims the benefit of priority under 35 U.S.C. 119(e)
to the filing date of *Carter, et al.*, U.S. provisional patent application
number 60/425,922 entitled "PROVISIONABLE PACKETCABLE
SUBSETS FOR EMBEDDED MULTIMEDIA ADAPTORS", which was
filed November 12, 2002, and is incorporated herein by reference in its
20 entirety.

FIELD OF THE INVENTION

 The present invention relates generally to providing telephony over
Internet protocol, and, more particularly, to facilitating the configuration of
25 interfacing features and capabilities of a cable modem with other network
devices.

BACKGROUND

As the use of digital data for transporting communication signals continues to grow in the consumer sector, more and more homes and offices are beginning to receive telephony services using an Internet connection. Furthermore, the physical network that transports the data may comprise a community antenna television (“CATV”) coaxial cable (“coax”) network. In such a scenario, a drop from a service provider’s outdoor coaxial cable plant connects to subscriber premise equipment (“SPE”), which is sometimes located outside a home or small office, or inside the premises. Within the SPE, television video signals and data signals, such as for example, Internet signals, are broken out and routed to their corresponding equipment. A cable modem is an example of an SPE with a coaxial cable connection for interfacing with a CATV network and typically an Ethernet or USB connection for transporting a data signal. To the cable modem, a media terminal adapter (“MTA”) is typically connected if a customer receives telephony services via the data signal. In addition, some manufacturers house the cable modem and MTA as a single device, for example, a TouchstoneTM Telephony Modem product (“TTM”), as offered by ARRIS International, Inc. Such an arrangement is typically referred to in the art as an embedded MTA (“EMTA”).

A TTM, for example, provides a user with telephony services over an Internet Protocol (“IP”) data network, which may utilize a cable modem termination system (“CMTS”). Cable telephony generally, the technology related to which is known in the art, provides a subscriber with telephony service that is transparent as to the source of the service. In other words, a

user plugs a telephone into a TTM and can access a set of features such as, for example, dial tone, call waiting and other features similar to those provided by a traditional plain old telephone system (“POTS”), typically implemented using the public switched telephone network (“PSTN”).

5 To provide telephony, a cable modem typically interfaces with a CMTS and other network system equipment, such as, for example, PacketCable provisioning servers, media gateways and PacketCable call management servers. Since voice over IP is still a developing technology, many vendors have not fully implemented the full set of PacketCable
10 features, which are specified within the protocols for communicating among various network devices that implement voice over IP using the existing CATV infrastructure. To avoid communication errors between devices that support different feature sets, technicians typically customize the software embedded into a cable modem, or other network devices, so that the device
15 for which the software is being customized can provide data in a format that other devices to which it will be communicating understand and support.

For example, if a cable modem manufactured by vendor A is to be used in a service provider’s network having a CMTS manufactured by vendor B, and vendor B has not implemented all of the features and
20 specifications of PacketCable, then engineers and technicians for vendor A may have to customize the software for the cable modem. This would facilitate recognition of modem data by the CMTS. This would also facilitate recognition of CMTS data by the modem. The customization may include translating the data sent from one format to another, or where
25 certain features are not implemented by one device, but are by the other, conditioning the data so that it is either eliminated from the data packets

sent, or nulled to merely comprise a place holder so that the remaining data bits that are sent will be received in the expected sequence.

Since there are many vendors of devices that implement the PacketCable specification, with each potentially implementing different parts or features of the specification, the customizing process increases the cost because engineers and technicians must analyze the interface requirements of each combination of network devices that are to communicate with one another and modify the embedded software accordingly for a given installation combination. Thus, the device operating software is typically loaded into a network device at the factory before shipment to the end user, either a consumer, with respect to cable modems, or service providers, with respect to CMTS and another head end devices. Although the software can also be loaded remotely into a device after it has been placed in use in a network, this still requires that engineers and technicians analyze the device requirements and capabilities, customize the software at the source code level for a particular combination of devices and download the software to the device.

Therefore, there is a need in the art for a method and system providing for the convenient and inexpensive configuration of communication interface features so that cable modems and other network devices implementing the PacketCable specification can communicate with other PacketCable devices.

SUMMARY

A method and system provide for the convenient and inexpensive configuration of PacketCable network devices to facilitate communication

with other PacketCable network devices. The entire PacketCable specification parameters are divided into predetermined subsets, as it has been discovered that certain vendors, relative to a particular piece of equipment having a particular level of functionality, implement certain PacketCable features and do not implement others. The features that are or are not implemented by a given vendor generally fall into definable subsets. By categorizing the features into defined subsets, operating software of a device can be configured to enable or disable the features of an entire subset or group. Thus, instead of having to modify the software at the source code level for a particular installation, a technician can enable or disable a given subset, or group, of features by provisioning device's operating software to turn on or off a software 'switch' or 'switches.'

For example, a vendor A cable modem is to be used in a network controlled by a vendor B CMTS, where vendor B does not use the features that are categorized into a particular subset. The software in the cable modem can be easily provisioned or configured to set a software switch or switches to disable the subset of features not implemented in the vendor B CMTS. Thus, an engineer need not write special code at the source code level to work around the non-support of these features by the remote server, or other piece of equipment.

In addition to the ability to set a software switch or switches, a user interface provides graphical user interface software for easily editing the operating system software to turn on or off the selected switches before the software is loaded into one or more cable modems, for example. Thus, a technician may use a personal computer to edit software parameters before they are loaded into a network device.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 illustrates PacketCable specification grouped into subsets of the entire specification with the capability to enable or disable any combination of subsets.

FIG. 2 illustrates a system and flow diagram for enabling certain specification subsets.

FIG. 3 illustrates a user interface displayed on a personal computer for specifying communication parameter values.

DETAILED DESCRIPTION

As a preliminary matter, it will be readily understood by those persons skilled in the art that the present invention is susceptible of broad utility and application. Many methods, embodiments and adaptations of the present invention other than those herein described, as well as many variations, modifications, and equivalent arrangements, will be apparent from or reasonably suggested by the present invention and the following description thereof, without departing from the substance or scope of the present invention.

Accordingly, while the present invention has been described herein in detail in relation to preferred embodiments, it is to be understood that this disclosure is only illustrative and exemplary of the present invention and is made merely for the purposes of providing a full and enabling disclosure of the invention. The following disclosure is not intended nor is to be construed to limit the present invention or otherwise to exclude other embodiments, adaptations, variations, modifications and equivalent

arrangements, the present invention being limited only by the claims appended hereto and the equivalents thereof.

Turning now to the figures, FIG 1 illustrates that the PacketCable specification 2 may be divided into subsets 4A-n. Through trial and error, it
5 has been discovered that certain features of the specification 2 tend to be grouped together by equipment vendors with respect to what their equipment does and does not support and implement. This equipment may include an embedded MTA within cable modem 6, the term cable modem herein after being used to refer to the EMTA portion of the cable modem, a
10 CMTS 8, a provisioning server 10, a media gateway 12 and a call management server 14, among others. These components are known to those skilled in the art, so their function with respect to voice over IP will not be discussed further. These components are typically connected over a hybrid fiber coaxial network, also known in the art. In addition, other
15 networking infrastructure technologies may be used to connect voice over IP devices together, including, but not limited to, coaxial cable, fiber optic, twisted pair, or wireless.

The PacketCable specification seeks to standardize the protocols for connecting and providing telephone calls over an IP network using the Data
20 Over Cable Service Interface Specification DOCSIS standard. However, as discussed above, all equipment that purports to be PacketCable-compatible does not necessarily implement all of the PacketCable-specification feature set. Thus, in order for a PacketCable device to communicate with another PacketCable device, each device must know what to expect when sending
25 and/or receiving a data packet to/from the other device. Otherwise, the receiving device may become confused as to what the data is, since it is

typically in binary format, and perform erratically, or not at all, if the data structure of a received packet does not comport with what it expects.

Thus, when the operating software 18 is loaded into a cable modem for example – the same would apply for other PacketCable devices as well -,
5 certain of the groups, or subsets 4, of specification 2 can be configured to be enabled or disabled before operating software 18 is loaded. This may be accomplished by selecting certain subsets 4 in software 18 to enable or disable software switches 20 inside modem 6 that turn on or off certain predetermined subsets comprising typically type length value (“TLV”) variables. When software 18 has been loaded and is operational, and the
10 cable modem sends out a packet, only data corresponding to what the CMTS, or other network device, expects is transmitted. Therefore, if subset 4C relates to emergency, or 911, calling, for example, and call management server 14 does not support emergency calling, then parameters relating to emergency calling will not be provisioned into the software load 18 of cable
15 modem 6 and software switch 20C corresponding to the emergency calling parameters will be turned off, as shown in the figure. Accordingly, a packet received by CMTS 8 and passed to management server 14 will only contain the information that the management server expects to receive and not parameters related to emergency calling. Computer 22 is used to select
20 parameter subsets and enter values there for.

Turning now to FIG. 2, a system 24 is shown for selecting and updating the operating system software for cable modem 6, having an EMTA. It will be appreciated that the same system can be used to select,
25 enter and load software parameters for other networks devices as well, including, but not limited to, provisioning server 10, CMTS 8, media

gateway 12 and call manager server 14. Process steps are shown with encircled numerals in the figure, and will be used to sequentially describe the method and system for implementing the loading and operation of cable modem's 6 software 18.

5 At step 1, a user, typically a engineer or technician working for a service provider, such as, for example, a cable television company, creates or edits a configuration file 26 using user interface 28, which may be a software based editor application running on PC 22. Based on the vendor of a given piece of equipment to which the cable modem 6 must communicate, 10 the technician may select the vendor and type of the equipment. Based on this input, the editor software may have prestored groupings of parameter subsets corresponding to the selected equipment that are displayed in parameters pane 30. If the prestored parameter information needs to be updated, the technician may edit the displayed parameters and values 15 thereof, using any of the control items shown on the right side of the interface. The technician may also use typical PC commands, such as cut, paste, overwrite and insert, for example, to edit the parameters directly in pane 30. Computer 22 may be located anywhere there is a network connection to CMTS 8 as shown by dashed line 32, but this will typically be 20 at the service provider head end or the administrative office.

When the configuration file 26 has been properly created/edited, it is transferred at step 2 to server 34. Server 34 is typically located at the head end and is typically a trivial file transfer protocol server ("TFTP"). From server 34, configuration file 26 is sent to cable modem 6, or other network 25 device being configured, at step 3. When file 26 is received at modem 6, the values of the variables within selected parameter subsets are used to set

software switches 20 inside the modem at step 4. These software switches 20 are typically implemented in the EMTA portion of modem 6, but may be implemented in a separate MTA device that is separate from the DOCSIS modem circuitry.

5 Switches 20 typically correspond to each of the plurality of parameter variables that exist in the PacketCable specification. Thus, in the figure, switch 20A, for example, controls a group, or subset, of parameter variables, and thus represents a software means for switching on or off more than one parameter. However, a subset is represented by one switch for clarity and to
10 show that certain parameters tend to be associated with others, so that for ease of setting up and editing of configuration file 26, one subset can be selected to simultaneously enable or disable all parameter variables in the given subset. However, it will be appreciated that if the parameters that a particular vendor's equipment recognizes do not match exactly with those in
15 the prestored groupings, or subsets, the technician can override the prestored groupings and enable or disable individual parameter variables within the groupings represented by switches 20, by using editor 28 before storing the selected information to file 26.

When file 26 has been received and switches 20 are configured
20 (turned on or off) to correspond to file 26, modem 6 configures its interface with the other network devices based on the switch settings, so that when packets are sent from the modem to the other devices – the modem typically communicates directly with CMTS 8, but indirectly through the CMTS with the other devices – the packets contain information in a format that is
25 recognizable by the receiving device, and thus communication errors are avoided. Moreover, when a software 'switch' 20 is off, certain pieces of

information, which may be entire packets of data, associated with the switch are not sent, for instance, from cable modem 6 to a particular server or other piece of equipment. For example, if a piece of equipment to which modem 6 will be communicating only supports single MAC functionality, at startup, the modem would not send DNS requests, AS, TGS and AP requests. In addition, modem 6 would not conduct SNMP messaging related to requesting and receiving MTA device capabilities and various file transfer aspects. Other features that are turned off are related to TFTP server addressing and sending of telephony provisioning information.

These features are known in the art and are given as examples of features that make up a particular subset related to single MAC functionality. The suppression of these requests and messaging related to certain features, which are grouped together as a subset, is controlled by variables that are included in the configuration file 26. The requests and other information sent from modem 6 are typically contained in separate packets, although more than one may be contained in the same packet. Thus, if the configuration file 26 instructs modem 6 that the equipment with which it will be communicating at startup only supports single MAC functionality, the modem will suppress, or 'switch off', the sending of packets containing data for setting up these features between the modem and the remote piece or piece or pieces of equipment

Turning now to FIG. 3, the preferred embodiment of the configuration editor user interface 28 is shown. Pane 30 shows variables for which values are used to control communication between network devices. For example, subset variable 36, 'arrisCmDevProvMethodIndicator.0', which corresponds to the provisioning method to be used, can be set to one

of several different values 38. In the figure, value 38 is set to 'singleMAC', which corresponds to the example given above in reference to FIG. 2. Using interface 28 to edit configuration file 26, when value 38 shown in FIG 3 is set to 'singleMAC', the subset of parameters discussed above
5 relating to single MAC functionality is indicated. When cable modem 6 referenced in FIG. 2 starts up with the configuration file set as such, software switches in the modem will be set to suppress the sending of packet data related to the non-supported features as indicated by value 38. Other subsets that may be indicated by entering different values 38 into
10 variable 36 include no suppression of features, which would indicate that the entire PacketCable specification is to be used. In addition, value 38 can be set to suppress features related to lack of support of KDC or to support GUPI functionality. These subsets, or groupings, of parameters to suppress sending of certain protocol information represent common patterns based on
15 current equipment being sold by various vendors. However, it will be appreciated that as more equipment is produced and offered for sale, other groupings of parameters into subsets can be made available to accommodate future products.

Similarly, when variable 40 is set to one of several different values
20 42, features that can be enabled/disabled include, but are not limited to, NCS message piggybacking, lockstep mode for the endpoint, transmission of wildcard NCS messages, RFC2833 messaging support, DSx-QoS or full PacketCable DQoS, provisional responses, multiple connections per line for supporting 3WC and CWT functionality, media encryption support, and
25 support for NCS error codes 404 and 540 for interoperation with call management servers. These features are part of the entire PacketCable

specification and are known to those skilled in the art. As shown in FIG. 3, value 42 is a numerical field, but those skilled in the art will appreciate that any form of variable or value can be used to convey configuration information from the configuration file to the device being configured.

5 When a given configuration file has been edited using editor 28, upon saving the configuration file using standard computer interface functions, such as, for example, selecting from a drop down menu 'save' or 'save as', computer 22 saves the file to server 34 in a format that is recognizable by the device being configured, as described above in reference to FIG. 2.

10 These and many other objects and advantages will be readily apparent to one skilled in the art from the foregoing specification when read in conjunction with the appended drawings. It is to be understood that the embodiments herein illustrated are examples only, and that the scope of the invention is to be defined solely by the claims when accorded a full range of
15 equivalents.